2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

Active Metric-Semantic Mapping by Multiple Aerial Robots

Xu Liu, Ankit Prabhu, Fernando Cladera, Ian D. Miller, Lifeng Zhou, Camillo J. Taylor, Vijay Kumar

***Abstract*— Traditional approaches for active mapping focus on building geometric maps. For most real-world applications, however, actionable information is related to semantically meaningful objects in the environment. We propose an approach to the active metric-semantic mapping problem that enables multiple heterogeneous robots to collaboratively build a map of the environment. The robots actively explore to minimize the uncertainties in both semantic (object classification) and geometric (object modeling) information. We represent the environment using informative but sparse object models, each consisting of a basic shape and a semantic class label, and characterize uncertainties empirically using a large amount of real-world data. Given a prior map, we use this model to select actions for each robot to minimize uncertainties. The performance of our algorithm is demonstrated through multi-robot experiments in diverse real-world environments. The proposed framework is applicable to a wide range of real-world problems, such as precision agriculture, infrastructure inspection, and asset mapping in factories.**

I. Introduction

Robots that can perceive and understand both semantic (e.g. class, species) and metric (e.g. shape, dimension) aspects of the environment and actively build metric-semantic maps can have a huge impact in many real-world applications. An example of our system is shown in Fig. 1.

Modeling the environment using a set of semantically meaningful object models is important for long-range exploration and large-scale mapping tasks. Such a model stores actionable information about the environment critical for robot exploration. In addition, semantic maps can provide long-term localization constraints for robot teams (loop closure, intra-robot registration) due to their viewpoint invariance. Finally, sparse semantic representations can significantly reduce the storage requirements, and are suitable for multi-robot settings with limited robot-to-robot communication bandwidth. Due to these motivations and advancements in data-driven object detection, metric-semantic Simultaneous Localization and Mapping (SLAM) methods are gradually becoming the new state-of-the-art in SLAM.

Despite the success of metric-semantic SLAM methods in various environments [1], [2], [3], [4], [5], [6], [7], active metric-semantic mapping remains an open and challenging research problem [8]. This problem requires robots to infer changes not only in geometric but also in semantic uncertainties as a result of their actions. The measurement model is complex and varies with viewing angle, distance, occlusions, robot motion, and object surface properties. Also, the noise models for geometric maps are not valid for semantic objects. Furthermore, characterization of uncertainties for semantic object classification and modeling is a challenging task [9].

Fig. 1: (Top left) Low-altitude UAV for map refinement. (Top right) high-altitude UAV for aerial mapping. (Bottom) Active metric-semantic mapping by two autonomous UAVs without using GPS.

In this paper, we propose a novel approach to the active metric-semantic mapping problem. Instead of making assumptions about the measurement model uncertainty or using heuristics for exploration, our algorithm empirically characterizes the noise from real-world observations. This can be intuitively thought of as allowing the robot to infer the uncertainty distributions based on its past observations of the world. The **contributions** of this paper include:

1) We propose a real-time metric-semantic SLAM algorithm that uses a generic, storage-efficient, semantically meaningful, and geometrically accurate environment representation, encodes object-robot constraints via customized factors in the factor graph to minimize robot odometry drift, and supports multi-robot collaboration.
2) We propose an active metric-semantic mapping algorithm built on the foundation of our metric-semantic SLAM algorithm. This algorithm uses empirical uncertainty characterizations from real-world data.
3) We integrate these algorithms with a complete autonomy stack and perform experiments in various real-world environments, including merging maps from multiple UAVs. The system is quantitatively demonstrated to gather higher quality information about objects of interest compared to benchmark methods.

To our knowledge, this work is the first to propose an active metric-semantic mapping system that enables the robot to minimize both metric and semantic uncertainties and is grounded in a systematic and empirical uncertainty characterization from a large amount of real-world data. A demo video of our system can be found at https://youtu.be/S86SgXi54oU.

This work was supported by funding from the IoT4Ag ERC funded by the National Science Foundation (NSF) under NSF Cooperative Agreement Number EEC-1941529, NIFA grant 2022-67021-36856, NSF grant CCR-2112665, and C-BRIC, a Semiconductor Research Corporation Joint University Microelectronics Program cosponsored by DARPA. Ian D. Miller acknowledges the support of a NASA Space Technology Research Fellowship. We gratefully acknowledge Alex Zhou for the hardware support, and Guilherme V. Nardari for contributing to the factor graph implementation.

X. Liu, A. Prabhu, F. Cladera, I. D. Miller, C. J. Taylor, V. Kumar are with GRASP Laboratory, University of Pennsylvania {liuxu, praankit, fclad, iandm, cjtaylor, kumar}@seas.upenn.edu.

L. Zhou was with GRASP Laboratory, University of Pennsylvania while completing this work. Presently, he is with the Department of Electrical and Computer Engineering, Drexel University lz457@drexel.edu.

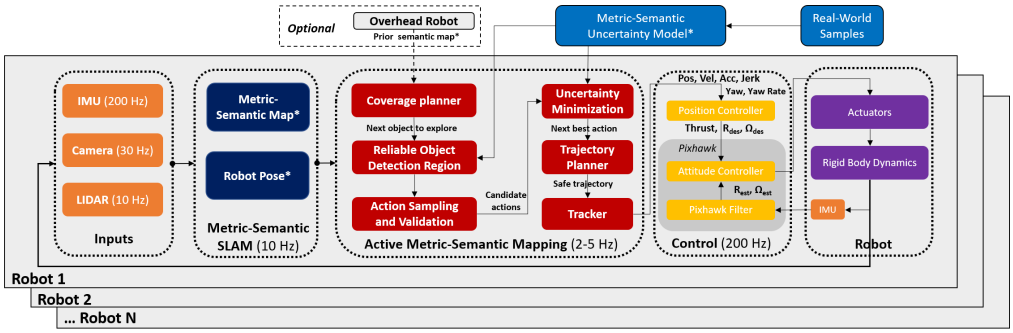

Fig. 2: **System architecture**. The quantities shared between robots are marked by asterisks, including the metric-semantic map (object models and labels), robot poses, semantic and geometric uncertainty distributions, and the prior semantic map from an overhead robot.

II. Related Work

We begin by discussing prior work in metric-semantic mapping, then active mapping, and finally works combining these areas into active metric-semantic mapping.

A. Metric-semantic Mapping

Many works in metric-semantic mapping represent the environment using dense semantically-annotated maps such as mesh [5], surfel [6], or 2.5D grid maps [10]. These maps can be conveniently used for planning and navigation but have large demands on computation and storage. It is challenging to use these approaches for large-scale mapping and exploration due to limited onboard computation. These works also do not employ semantics to improve localization accuracy. Additionally, individual object models, often desirable for downstream tasks such as asset mapping, are not used in the SLAM optimization. By contrast, other works build object-level maps using pre-collected 3D models [11], or basic shapes such as points [1], cuboids [4], ellipsoids [2], cylinders [7], a set of semantic keypoints [12], or a combination of 2D shapes [13]. Some efforts also account for ambiguity in data association [1], and the usage of semantic information in identifying moving objects [14], generating hierarchical descriptors for loop closure [15], and minimizing odometry drifts to assist long-range navigation [16]. However, these approaches do not define a measurement model that maps from state space to object classification confidence space, which is still an open and challenging problem [9]. Without such a model, these approaches cannot be adopted for active information acquisition in metric-semantic maps, where the robot needs to infer how semantic and geometric uncertainties are affected by its actions.

B. Active mapping

Prior works on active mapping mostly use geometric maps, which represent the environment using either sparse (e.g. landmark-based) or dense (e.g. volumetric) elements. [17] proposes an efficient active information acquisition algorithm for sparse maps. [18] extends this work by decentralizing it for multiple robots, and adaptively adjusting the sub-optimality to satisfy the computation budget. For dense maps, [19] proposes an information-theoretic active 3D occupancy grid mapping algorithm heavily optimized to run in real time. To explore larger environments, some works employ a two-stage approach where the algorithm first plans paths for coverage and then refines the path for maximizing information gain or minimizing execution time [20], [21]. However, none of these methods account for uncertainties in semantic information, which we show often has a significantly different distribution than geometric uncertainties.

C. Active Metric-semantic mapping

Some prior works seek to characterize the uncertainty related to semantic objects [22], [23], but cannot predict uncertainties of future measurements. Others address this next-best viewpoint prediction problem [24], [25] but verify the models only under controlled indoor conditions with prior knowledge, such as detailed 3D object models. Finally, some works take a more end-to-end approach using reinforcement learning [26] or map prediction [27], but validate their algorithms only in simulation. Others take a more model-based approach using Gaussian Mixture Models [28] or Bayesian OcTrees [29], [30] to model uncertainties. However, none of them explicitly model the relationship between object classification uncertainty and states of robot and object.

We approach the active metric-semantic mapping problem using model-based information-theoretic exploration, where the uncertainties of the metric-semantic measurement model are characterized empirically based on a large amount of real-world data. We leverage our semantic SLAM module to automate this characterization process. Our approach allows robots to explore and build an uncertainty-minimized metric-semantic map in real time.

III. Preliminaries and Problem Formulation

A. Preliminaries

Let there be k robots $\{r^1, r^2, ..., r^k\}$. The semantic map of the kth robot $\mathcal{M}^k$ consists of a set of semantic objects $\{\ell_1^k, \ell_2^k, ..., \ell_n^k\}$ belonging to v semantic classes $\{s_1, s_2, ..., s_v\} \in \mathcal{S}$. For notational compactness, we suppress k unless otherwise noted. Each object has a state vector $\ell_i = \{\ell_i^s, \ell_i^g\}$ that defines the semantic and geometric properties of the object, where $\ell_i^s = (p_i, s_i) \in [0, 1] \times \mathcal{S}$, p_i is the probability of ℓ_i belonging to class s_i, and ℓ_i^g is a vector defining the pose and geometric model of the object. The semantic class of the object defines the shape of the object, which is rectangular cuboid, cylinder, or plane[1]. For each cuboidal object, the state vector is (suppressing subscript i):

[1]Planar objects are associated with ground classes and are implicitly used to constrain the bottom of the cylinder and cuboid models.

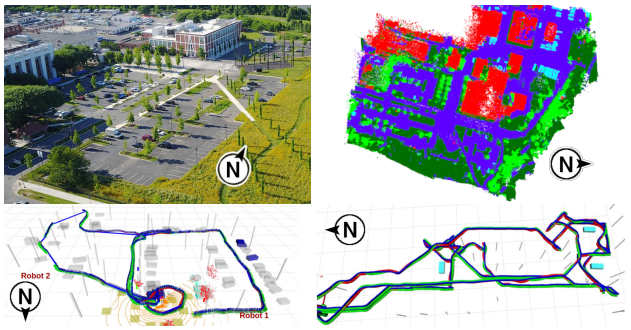

Fig. 3: **Active semantic mapping of urban areas with a heterogeneous team of UAVs**. (Top left) Overhead view of one of our experiment environments. (Top right) Semantic map built by the overhead robot in real time, where vehicles are cyan colored. (Bottom) Examples of the semantic map built by two low-altitude robots. Cylinders represent light poles or tree trunks, cuboids represent vehicles, and planes represent the local ground.

$\ell^g = [\mathbf{r}; \mathbf{t}; \mathbf{d}]$, where $\mathbf{r} = [r_x, r_y, r_z]^\intercal$ is the rotation vector, $\mathbf{t} = [t_x, t_y, t_z]^\intercal$ is the translation vector, and $\mathbf{d} = [d_x, d_y, d_z]^\intercal$ is the dimension vector. For each cylindrical object, the state vector is: $\ell^g = [\mathbf{b}; \mathbf{n}; r]$, where $\mathbf{b} = [b_x, b_y, b_z]^\intercal$ is the origin of the axis ray, $\mathbf{n} = [n_x, n_y, n_z]^\intercal$ is the direction of the axis ray, and r is the radius. The robot state at time t is represented by $\mathbf{x}_t$, which contains the $\mathbb{SE}(3)$ pose.

We assume that the base (take-off) locations of robots are close and known relative to each other, and allow robots to communicate and exchange information only at the base. Note that this implies that the robots explore independently, but can collaboratively construct semantic maps when at the base. Also, simultaneous and sequential flights are equivalent under this assumption. In our prior work [31], we proposed a method for localizing robots using a semantic map built by a high-altitude UAV and opportunistically communicating data within the team. Using these methods, we can relax our assumptions and extend our work to handle intermittent communication or unknown take-off positions.

B. Objective

Given an unknown environment, or limited prior knowledge about the environment, our objective is to find $\hat{\mathcal{M}}$, such that $\|\hat{\mathcal{M}} \ominus \mathcal{M}\|$ is minimized. $\ominus$ is the generalized difference between the two maps. In other words, we want our estimated metric-semantic map to best approximate the real-world map. We decompose our problem into metric-semantic SLAM and Planning for Active Metric and Semantic information acquisition (PAMS).

1) Metric-Semantic SLAM: The objective of metric-semantic SLAM is to accumulate previous measurements $\{\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_t\} \in \mathcal{Z}_t$ to estimate the current metric-semantic map $\mathcal{M}_t$ and a set of robot trajectories $\{\mathcal{T}_1, \mathcal{T}_2, .., \mathcal{T}_k\}$.

2) PAMS: The objective of PAMS is to find the best trajectory for each robot that minimizes both **semantic** and **geometric** uncertainties, given $\mathcal{Z}_t$ and the initial states of all robots $\mathbf{x}_0$, and a planning horizon $\boldsymbol{\tau} \triangleq t+1 : t+T$. Recall that robots explore independently, so we treat PAMS as a single-robot problem. Minimizing semantic and geometric uncertainty requires confidently classifying an object, and reducing errors in geometric models of objects ℓ_i^g. Instead of directly optimizing the two objectives together, we cast

Fig. 4: **Experiment environments** with vehicles, light poles, tree trunks. (Left) Garage. (Middle Two) Urban. (Right) Dirt road.

the classification confidences p_i as the constraint of the optimization problem, and minimizing the uncertainties of ℓ_i^g as the objective[2]. In other words, we try to improve the accuracy of the object model, once the object exists and belongs to the class of interest. This design avoids hand-tuning the weights of different objectives and improves computational efficiency and optimization convergence.

IV. Proposed Approach

A. Object detection and modeling

For each object geometry type, we construct a virtual sensor, which given raw point cloud data outputs the estimated object configuration measurements in the body frame. This pipeline has three steps: semantic segmentation, instance extraction, and object modeling.

For semantic segmentation, we build upon RangeNet++ [32], and drastically reduce the number of layers in the encoder and decoder to boost efficiency ($\sim$800% the speed of the default DarkNet-53 [33] backbone). It runs in real time on the Intel NUC computer.

The extraction and modeling of cylindrical objects and local ground planes are done in a similar way as presented in our previous work [7], and the root $\mathbf{b}$ of the cylinder model is the intersection of cylinder axis and local ground plane. For cylindrical objects, the resulting measurement is $\ell^g(\mathbf{z}) = [\mathbf{b}(\mathbf{z}); \mathbf{n}(\mathbf{z}); r(\mathbf{z})]$, but in robot body frame.

For cuboidal objects, the resulting measurement is of the form $\ell^g(\mathbf{z}) = [\mathbf{r}(\mathbf{z}); \mathbf{t}(\mathbf{z}); \mathbf{d}(\mathbf{z})]$, but again in robot body frame. Cuboidal objects are difficult to model from only one LIDAR scan. Therefore, we use LIDAR odometry to accumulate a 1$\sim$3 seconds of semantically segmented point clouds and filter points into a certain elevation window to reject outlier points. Filtered points with target class labels are clustered using DBSCAN [34]. We then project the points within each cluster onto the ground plane, and extract the 2D convex hull of each cluster. We perform principal component analysis (PCA) on the 2D convex hull points to estimate the longitudinal axis $\mathbf{b}_1$ of the cuboid (first PCA component). We assume that the front of the vehicle is lower than the rear to determine the facing direction. The vertical axis $\mathbf{b}_3$ of the cuboid is assumed to be the same as the normal of its nearby local ground plane. The lateral axis is then $\mathbf{b}_2 = \mathbf{b}_3 \times \mathbf{b}_1$. The dimensions $\mathbf{d}(\mathbf{z}) = [d_x, d_y, d_z]^\intercal$ are estimated by the distance between the 5th and 95th percentiles of projections onto the axes, and $\mathbf{b}_2$, $\mathbf{b}_3$ define the rotation $\mathbf{r}(\mathbf{z})$. The centers of the projections define the translation $\mathbf{t}(\mathbf{z})$.

B. Metric-Semantic SLAM

We build our semantic SLAM implementation on the GTSAM library [35], [36], [37]. The factor graph of our

[2]We assume that the robot pose estimation error is negligible, since we use an accurate LIDAR-inertial odometry algorithm to estimate frame-to-frame relative transformation and semantic landmarks to minimize the drift.

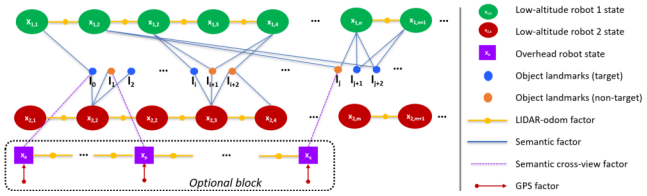

Fig. 5: Metric-semantic SLAM factor graph representation. We define customized factors for cuboidal and cylindrical objects.

semantic SLAM module is shown in Fig. 5. The objects are classified into target and non-target objects, which are discussed in Section IV-D.

Let $\mathbf{H}_s^w$ be the matrix form of the robot pose $\mathbf{x}_t$, where $\mathbf{R}_s^w$ is the rotation component and $\mathbf{t}_s^w$ is the translational component. We suppress the i subscript of ℓ_i for notational compactness. Following [35], [36], the measurement likelihood function with the Gaussian noise model is: $\mathbf{L}(\mathbf{x}, \ell^g; \mathbf{z}^g) = \mathbf{exp}\{-\frac{1}{2}||\mathbf{h}(\mathbf{x}, \ell^g) \ominus \mathbf{z}^g||^2_\Sigma\}$, where $\mathbf{h}(\mathbf{x}, \ell^g) \ominus \mathbf{z}^g$ is the error $\mathbf{e}_{(\cdot)}$ that we will define separately for cuboid and cylinder objects, $\mathbf{z}^g$ is the output of the object detection and modeling step, and Σ is the covariance matrix.

1) Odometry factor: We use a LIDAR odometry algorithm [38] to generate the odometry factor. We calculate a relative $\mathbb{SE}(3)$ transform between two consecutive pose estimates and use this transform as a factor between the corresponding poses in the graph.

2) Data association: When detecting an object, we first associate it with objects in $\mathcal{M}$ or create a new map object. We employ nearest neighbor (NN) matching of the centroids (for cuboids) or roots (for cylinders), with a fixed threshold for valid matches, using the currently estimated robot pose to transform detected objects into the world frame for matching.

3) Custom cuboid and cylinder factors: Once we have object associations, we use these observation-landmark matches to form factors in the factor graph. To acquire the expected measurement of the cuboid, we can first transform the pose of the cuboid into the robot body frame $\mathbf{H}^s_{\text{cub}} = \mathbf{H}^s_w \mathbf{H}^w_{\text{cub}}$. We use a similar measurement error function as the one in [4] for cuboidal objects, i.e.,

$$\mathbf{e}_{\text{cub}} = \begin{bmatrix} \mathbf{log}((\mathbf{H}^s_{\text{cub}}(\mathbf{z}))^{-1}(\mathbf{H}^s_w \mathbf{H}^w_{\text{cub}}))^\vee \\ \mathbf{d} - \mathbf{d}(\mathbf{z}) \end{bmatrix} \tag{1}$$

where $\vee$ is vee operator that maps the $\mathbb{SE}(3)$ transformation matrix into 6×1 vector, $\mathbf{log}$ is the log map, and $(\cdot)(\mathbf{z})$ are the object measurements. Intuitively, this measurement error is the distance between the currently detected and the expected cuboid models in the tangent space of $\mathbb{SE}(3)$ and the 3×1 dimension vector.

Similarly, we can calculate the expected measurement and actual measurement of cylinder objects from ℓ^g_i and $\ell^g_i(\mathbf{z})$. We define the measurement error function for cylindrical objects as:

$$\mathbf{e}_{\text{cyl}} = \begin{bmatrix} (\mathbf{R}^s_w \mathbf{b} + \mathbf{t}^s_w) - \mathbf{b}(\mathbf{z}) \\ \mathbf{R}^s_w \mathbf{n} - \mathbf{n}(\mathbf{z}) \\ \mathbf{r} - \mathbf{r}(\mathbf{z}) \end{bmatrix} \tag{2}$$

We implemented our custom factors to be compatible with the GTSAM library [35].

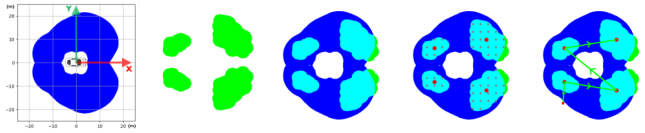

Fig. 6: For a vehicle at (0,0) facing in the +x direction: (1st) Region with 95% confidence in semantic classification. (2nd) Region with 95% confidence in localization and dimension estimation (with less than 0.1 m and 0.2 m error, respectively). (3rd) Regions with 95% confidence level for both semantic and metric mapping are marked in cyan. (4th) Sampling. Two different sampling strategies: uniform sampling (grey) and centroid-only sampling (red), which form the set of valid samples, i.e., $\mathcal{X}^{\texttt{visible}}$. (5th) Best action sequence.

4) Multi-robot Semantic SLAM: Each robot maintains its own factor graph. When robots establish communication, they exchange their history object detections and odometry, with each other. Robots can treat other robots' data in the same way as their own by maintaining an odometry graph for each robot and performing data association in the same way as their own measurements. This merged factor graph is illustrated in Fig. 5.

C. Uncertainty modeling

As discussed in Section I, we seek to develop a pipeline for building models that map from the state space to the metric-semantic uncertainty space from real-world data. Formally, we want to find f^s and f^g such that $p_i = f^s(\mathbf{x}, \ell^g_i, \ell^s_i), \mathbf{\Sigma}(\ell^g_i) = f^g(\mathbf{x}, \ell^g_i, \ell^s_i)$, where f^s maps state space into object classification confidence space, and f^g maps state space into object geometric measurement uncertainty space.

1) Acquiring the training samples: To characterize the uncertainty, we propose extracting important low-dimensional inputs to reduce the dimensionality of the problem. From our data, we found that the most influential factors for object classification and modeling accuracies are range and viewing angles. Thus, we use these two quantities as inputs to the map predicting the uncertainty. To generate data, we manually fly a spiral-shaped trajectory centered around the object of interest to cover as many angles and ranges as possible.

In order to model uncertainty, we require pairs of observation and ground truth to fit the model. To characterize the object classification uncertainty, we first extract the points enclosed by the cuboid or cylinder model estimated by our semantic mapping algorithm from the recent 10∼30 time window. We then calculate the average per-point classification confidence from our semantic segmentation network, and use this as the object classification confidence. Once these samples are collected, we use a multilayer perceptron to approximate the underlying semantic uncertainty distribution and can then predict the confidence for any range and viewing angle, as shown in Fig. 6. A visualization is here.

To obtain geometric model ground truth, we look up the actual dimensions of the vehicle online. To characterize geometric uncertainty, we first discretize the range and viewing angle. For each discretized interval, we calculate the error distribution of all measurements in the interval compared to the ground truth object model. A spline is fit to these data points which represent the underlying geometric uncertain-

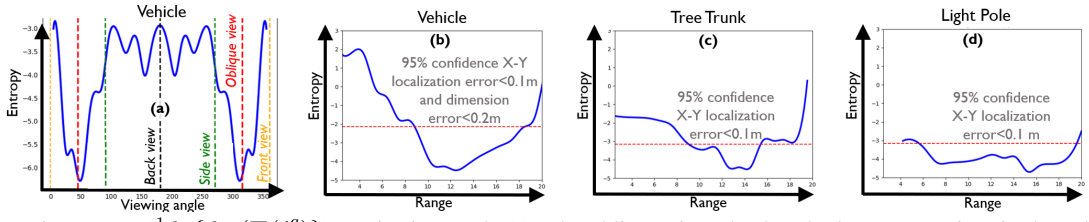

Fig. 7: Geometric entropy ($\frac{1}{2}\ln\{\det(\mathbf{\Sigma}(\ell_i^g)\}$) vs viewing angle (a). The oblique views lead to the least uncertainty in the vehicle geometric model. Geometric entropy vs range (b-d). For each semantic class, there is a range interval that leads to minimized geometric uncertainty.

ties. The results of this characterization are shown in Fig. 7. Note that for vehicles we only consider uncertainties in X-Y localization, length, and width, since the Z position is well constrained by the ground, and height is observable from all viewing angles. For light poles and tree trunks, we only consider uncertainties in X-Y localization.

D. PAMS

The semantic classes are divided into target objects (vehicles), which are used to guide active semantic mapping, and non-target objects (tree trunks and light poles), which are only used to minimize robot localization drift.

1) Target object discovery: With the object of interest locations extracted from the prior semantic map, we simply solve or approximate the Traveling Salesman Problem (TSP) to visit all of the objects as efficiently as possible. Our implementation also supports reactive exploration (active mapping upon detecting an object of interest), and it is trivial to incorporate a coverage planner into our system [16].

2) Uncertainty minimization: As presented in Section III, once the robot detects a target object it will minimize uncertainties in the geometric properties for the target object, while guaranteeing that the classification confidence and robot localization uncertainty are within a threshold.

To guarantee object classification confidence, candidate actions are only sampled from the high confidence regions. To further reduce the sampling space, we also extract the low geometric uncertainty regions and samples in the intersection of the two, either by uniform sampling or by taking their centroids, as illustrated in Fig. 6.

We then generate candidate paths given the planning horizon T constituting all possible orders of visiting the a sample locations. Therefore, the number of candidate actions is equal to the number of variations $P(a,T) = a!/(a-T)!$.

For each candidate path $\mathbf{x}_\tau = [\mathbf{x}_{t+1}, \mathbf{x}_{t+1}, ...\mathbf{x}_{t+\tau}]$, we quantify the information gain. Supposing that ℓ_i^g is the target object to explore for time t, we use differential entropy to calculate the information gain as: $\mathbf{I}(\ell_i^g; \mathbf{z}_\tau) = \frac{1}{2}\ln\{\det(\mathbf{\Sigma}(\ell_{i,t}^g))\} - \frac{1}{2}\ln\{\det(\mathbf{\Sigma}(\ell_{i,t+T}^g))\}$. We update the covariance recursively using the same method as shown in [17]: $\mathbf{\Sigma}(\ell_{i,t+1}^g) = (\mathbf{\Sigma}(\ell_{i,t}^g)^{-1} + \mathbf{\Sigma}(\ell_{i,t+1}^g)^{-1})^{-1}$, where $\mathbf{\Sigma}(\ell_{i,t+1}^g)$ is estimated by the uncertainty models we obtained in Section IV-C. We choose the candidate path maximizing this information gain, and end exploration once entropy drops below a threshold or all samples are explored.

V. Results and Analysis

To demonstrate the performance of our system, we carried out experiments in various real-world environments as shown in Fig. 4 which are diverse in object types, object shapes, and degree of structure. As illustrated in Fig. 1, our complete system consists of both low-altitude and overhead UAVs, both built on our custom-made Falcon 4 UAV platform [16]. The low-altitude UAVs are equipped with forward-facing cameras and 3D LIDARs, and can autonomously navigate in cluttered and GPS-denied environments. To focus on the map quality, we assume that there is a prior map available (which could come from the high-altitude UAV) and the robots only explore the objects in this map. Our system detects and models vehicles (as cuboids), tree trunks and light poles (as cylinders), and the ground (as planes).

A. Qualitative results

We observe in Fig. 6 and Fig. 7 that both semantic and geometric uncertainties vary significantly for the vehicle class when either the angle or the range changes, and these trends are clear and consistent across multiple datasets. However, for the tree trunk and light pole classes, the semantic confidence does not have a clear relationship with range or viewing angle. Even when there are very few points returned from the object, the segmentation network can classify them relatively well. This is probably because these objects are relatively distinct from the surrounding environment. There is a clear relationship, though, between the range and the geometric uncertainty for light poles and tree trunks, as seen in Fig. 7. Based on these observations, we model the distribution of metric and semantic uncertainties w.r.t. both range and viewing angle for the vehicle class, while only modeling the distribution of metric uncertainties w.r.t. range for the light pole and tree trunk classes.

As illustrated in Fig. 6, the classification confidence is higher from the two side views than from the front or back views, while the back view is more informative than the front view. This distribution is intuitive since from the side view, the vehicle is more readily identified. We also note from Fig. 6 that the dimensions of vehicles are best estimated from oblique views. This is reasonable because, at oblique views, the sensor can observe both the lateral and longitudinal directions of the vehicle, resulting in a better fit.

In addition, our uncertainty characterization approach generalizes to different types of objects as in Fig. 7. The method can characterize uncertainties for any object of interest provided that the object shape can be approximated by cuboids or cylinders and the sensor used can output point cloud data.

In Fig. 8 we compare the trajectories from active semantic mapping to our baseline method of visiting each object location in the prior map. While one could likely design a heuristic to perform similarly to our method, we emphasize that our system is rooted in real-world data instead of human intuition, and therefore is more generalizable. Our system

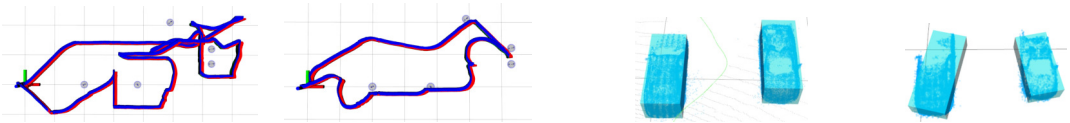

Fig. 8: **Comparison of trajectory** examples from active metric-semantic mapping (leftmost) and heuristic-based mapping (middle left). Each grid is 10 m × 10 m. The ground-truth vehicle positions are marked by blue-colored disc-shaped markers. Our active metric-semantic mapping method drives the robot to observe objects from different ranges and viewing angles to minimize uncertainties. **Comparison of estimated vehicle model** examples from active metric-semantic mapping (middle right) and heuristic-based mapping (rightmost).

Measure	Urban Dataset 1	Urban Dataset 2	Urban Dataset 3	Urban Dataset 4	Parking Garage	Dirt Road	**All Data**
Err. Mean. (Ours / Baseline)	**0.26 m** / 0.44 m	**0.22 m** / 0.64 m	**0.03 m** / 0.46 m	**0.04 m** / 0.71 m	**0.68 m** / 1.96 m	0.58 m / **0.39 m**	**0.19 m** / 0.57 m
Err. Std. Dev. (Ours / Baseline)	**0.26 m** / 0.35 m	**0.33 m** / 0.39 m	**0.30 m** / 0.47 m	0.36 m / **0.34 m**	—	0.20 m / **0.19 m**	**0.39 m** / 0.55 m
Obj. Mapped (Ours / Baseline)	100% / 100%	100% / 100%	100% / 100%	**100**% / 80%	100% / 100%	100% / 100%	**100**% / 95%

Fig. 9: Geometric model error mean and standard deviation and percentage of objects discovered.

achieves a 100% detection accuracy and is significantly better in terms of object model accuracy than the heuristic-based method as shown on the right of the figure. The higher accuracy does come at the cost of a longer path length, but the difference in path length is less significant when the environment is more sparse because the robot will spend proportionally more time covering the environment as opposed to exploring individual objects.

It is obvious that the metric-semantic uncertainty distributions are highly non-linear and vary with object types, as shown in Fig. 7. It is difficult to design and verify mathematical models for such uncertainty distributions, especially for complicated objects that vary in type and appearance.

B. *Quantitative results*

We evaluate our system quantitatively using the accuracy of width and length estimates for target-class objects (i.e. vehicles) as the metric. We looked up the dimensions based on the vehicle model to obtain the ground truth. For a discussion of localization error, we refer readers to our prior work [7], [16] which demonstrated the ability to reduce long-term drift in robot localization using semantic SLAM.

We compare against our baseline method as described in Section V-A. The object modeling accuracy is significantly improved using the proposed method, as shown in Fig. 9. The overall error mean is 0.19 m, which is around 3.5% of the average vehicle dimensions. The overall error standard deviation is 0.39 m, which means that the vehicle dimensions can be estimated within 14.5% error with more than 95% confidence. By comparison, across all datasets, the error mean and standard deviation of the heuristic-based baseline method are around 300% and 140% of the error mean and standard deviation of our method. This shows that, by actively choosing the best viewing angles and ranges, the robot can gather significantly more accurate information.

We also note that there are edge cases where our method struggles. For example, the parking garage is a constrained space where most of the exploration viewpoints are unreachable. The dirt road is highly unstructured, lacking reliable landmarks such as tree trunks or light poles to constrain the robot's pose. Therefore, the longer the robot operates, the more drift accumulates, and the less accurate the model becomes. Thus, the heuristic-based approach that requires shorter flights outputs better results in this case. Additionally, Urban Dataset 1 contains all black vehicles, which are generally more difficult to detect, thus resulting in more noise. A potential solution is accounting for LIDAR intensities in our uncertainty characterization models.

Measure	Robot at 2.5 m	Robot at 5 m	Combined
Num. of Obj. / Min	0.8	**0.9**	0.8
Err. Mean	0.13 m	0.15 m	**0.06 m**
Err. Std. Dev.	**0.30 m**	0.50 m	**0.30 m**

Fig. 10: Mapping accuracy vs the number of heterogeneous robots.

Measure	Robot in lot 1	Robot in lot 2	Robot in lot 3	Combined
Num. of Obj. / Min	0.8	1.6	1.6	**3.3**
Err. Mean	0.13 m	0.22 m	**0.03 m**	0.12 m
Err. Std. Dev.	**0.30 m**	0.33 m	**0.30 m**	0.33 m

Fig. 11: Mapping efficiency vs the number of homogeneous robots.

We perform multi-robot experiments by flying sequentially and fusing the maps afterwards. Using heterogeneous robots leads to more accurate mapping results than any robot working independently as in Fig. 10. Homogeneous robots can parallelize coverage linearly w.r.t. the number of robots without sacrificing map accuracy as in Fig. 11, assuming that the robots fly simultaneously.

Our full system runs in real time onboard UAVs with an Intel i7-10710U processor. Semantic segmentation runs at 1.5 Hz using only 1 out of 12 CPU threads. In total we use 5-6 threads, leaving ∼50% computational headroom.

Finally, our map representation significantly reduces storage and communication burden. One million object models take only 9 MB of storage. By comparison, it requires 6,250 MB to store a 3D voxel map that covers the same area (0.1 m resolution, 20 m height, 5 m spacing between objects).

VI. Conclusion and Future Work

Highly accurate metric-semantic maps are important for applications such as precision agriculture, infrastructure inspection, and asset mapping in factories. In this paper, we proposed an active metric-semantic mapping approach that enables robots to actively explore to minimize uncertainties in both metric and semantic information. We characterized the relationship between states and metric-semantic uncertainties empirically using a large amount of real-world data, and analyzed the resulting insights into the relationship between uncertainty and different viewpoints. Through real-world multi-robot experiments, we showed that our system is capable of constructing highly accurate metric-semantic maps in real time. We additionally showed that our active mapping system improved map quality, and using multiple heterogeneous robots can improve both the map accuracy as well as the mapping speed. Future work includes scaling up to include more semantic classes, enabling robots to react to each other's information in real time, and considering more factors such as altitude angle and LIDAR point cloud intensity for the uncertainty characterization.

REFERENCES

[1] S. L. Bowman, N. Atanasov, K. Daniilidis, and G. J. Pappas, "Probabilistic data association for semantic slam," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 1722–1729.

[2] L. Nicholson, M. Milford, and N. Sünderhauf, "Quadricslam: Dual quadrics from object detections as landmarks in object-oriented slam," *IEEE Robotics and Automation Letters*, vol. 4, no. 1, pp. 1–8, 2018.

[3] N. Atanasov, S. L. Bowman, K. Daniilidis, and G. J. Pappas, "A unifying view of geometry, semantics, and data association in slam." in *IJCAI*, 2018, pp. 5204–5208.

[4] S. Yang and S. Scherer, "Cubeslam: Monocular 3-d object slam," *IEEE Transactions on Robotics*, vol. 35, no. 4, pp. 925–938, 2019.

[5] A. Rosinol, A. Violette, M. Abate, N. Hughes, Y. Chang, J. Shi, A. Gupta, and L. Carlone, "Kimera: From slam to spatial perception with 3d dynamic scene graphs," *The International Journal of Robotics Research*, vol. 40, no. 12-14, pp. 1510–1546, 2021.

[6] X. Chen, A. Milioto, E. Palazzolo, P. Giguere, J. Behley, and C. Stachniss, "Suma++: Efficient lidar-based semantic slam," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 4530–4537.

[7] S. W. Chen, G. V. Nardari, E. S. Lee, C. Qu, X. Liu, R. A. F. Romero, and V. Kumar, "Sloam: Semantic lidar odometry and mapping for forest inventory," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 612–619, 2020.

[8] J. A. Placed, J. Strader, H. Carrillo, N. Atanasov, V. Indelman, L. Carlone, and J. A. Castellanos, "A survey on active simultaneous localization and mapping: State of the art and new frontiers," *planning*, vol. 2, 2022.

[9] D. M. Rosen, K. J. Doherty, A. Terán Espinoza, and J. J. Leonard, "Advances in inference and representation for simultaneous localization and mapping," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 4, pp. 215–242, 2021.

[10] D. Maturana, P.-W. Chou, M. Uenoyama, and S. Scherer, "Real-time semantic mapping for autonomous off-road navigation," in *Field and Service Robotics*. Springer, 2018, pp. 335–350.

[11] R. F. Salas-Moreno, R. A. Newcombe, H. Strasdat, P. H. Kelly, and A. J. Davison, "Slam++: Simultaneous localisation and mapping at the level of objects," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 1352–1359.

[12] M. Shan, Q. Feng, and N. Atanasov, "Orcvio: Object residual constrained visual-inertial odometry," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 5104–5111.

[13] B. Cao, R. C. Mendoza, A. Philipp, and D. Göhring, "Lidar-based object-level slam for autonomous vehicles," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 4397–4404.

[14] X. Chen, S. Li, B. Mersch, L. Wiesmann, J. Gall, J. Behley, and C. Stachniss, "Moving object segmentation in 3d lidar data: A learning-based approach exploiting sequential data," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 6529–6536, 2021.

[15] N. Hughes, Y. Chang, and L. Carlone, "Hydra: A real-time spatial perception engine for 3d scene graph construction and optimization," *arXiv preprint arXiv:2201.13360*, 2022.

[16] X. Liu, G. V. Nardari, F. C. Ojeda, Y. Tao, A. Zhou, T. Donnelly, C. Qu, S. W. Chen, R. A. F. Romero, C. J. Taylor, *et al.*, "Large-scale autonomous flight with real-time semantic slam under dense forest canopy," *IEEE Robotics and Automation Letters*, 2022.

[17] N. Atanasov, J. Le Ny, K. Daniilidis, and G. J. Pappas, "Information acquisition with sensing robots: Algorithms and error bounds," in *2014 IEEE International conference on robotics and automation (ICRA)*. IEEE, 2014, pp. 6447–6454.

[18] B. Schlotfeldt, D. Thakur, N. Atanasov, V. Kumar, and G. J. Pappas, "Anytime planning for decentralized multirobot active information gathering," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 1025–1032, 2018.

[19] B. Charrow, S. Liu, V. Kumar, and N. Michael, "Information-theoretic mapping using cauchy-schwarz quadratic mutual information," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 4791–4798.

[20] B. Charrow, G. Kahn, S. Patil, S. Liu, K. Goldberg, P. Abbeel, N. Michael, and V. Kumar, "Information-theoretic planning with trajectory optimization for dense 3d mapping." in *Robotics: Science and Systems*, vol. 11, 2015.

[21] B. Zhou, Y. Zhang, X. Chen, and S. Shen, "Fuel: Fast uav exploration using incremental frontier structure and hierarchical planning," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 779–786, 2021.

[22] H. Yu, J. Moon, and B. Lee, "A variational observation model of 3d object for probabilistic semantic slam," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 5866–5872.

[23] T. Cortinhal, G. Tzelepis, and E. Erdal Aksoy, "Salsanext: Fast, uncertainty-aware semantic segmentation of lidar point clouds," in *International Symposium on Visual Computing*. Springer, 2020, pp. 207–222.

[24] N. Atanasov, B. Sankaran, J. Le Ny, G. J. Pappas, and K. Daniilidis, "Nonmyopic view planning for active object classification and pose estimation," *IEEE Transactions on Robotics*, vol. 30, no. 5, pp. 1078–1090, 2014.

[25] A. Doumanoglou, R. Kouskouridas, S. Malassiotis, and T.-K. Kim, "Recovering 6d object pose and predicting next-best-view in the crowd," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3583–3592.

[26] D. S. Chaplot, M. Dalal, S. Gupta, J. Malik, and R. R. Salakhutdinov, "Seal: Self-supervised embodied active learning using exploration and 3d consistency," *Advances in Neural Information Processing Systems*, vol. 34, pp. 13 086–13 098, 2021.

[27] G. Georgakis, B. Bucher, K. Schmeckpeper, S. Singh, and K. Daniilidis, "Learning to map for active semantic goal navigation," *arXiv preprint arXiv:2106.15648*, 2021.

[28] C. Wang, J. Cheng, W. Chi, T. Yan, and M. Q.-H. Meng, "Semantic-aware informative path planning for efficient object search using mobile robot," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2019.

[29] A. Asgharivaskasi and N. Atanasov, "Active bayesian multi-class mapping from range and semantic segmentation observations," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 1–7.

[30] ——, "Active bayesian multi-class mapping from range and semantic segmentation observations," *CoRR*, vol. abs/2112.04063, 2021. [Online]. Available: https://arxiv.org/abs/2112.04063

[31] I. D. Miller, F. Cladera, T. Smith, C. J. Taylor, and V. Kumar, "Stronger together: Air-ground robotic collaboration using semantics," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9643–9650, 2022.

[32] A. Milioto, I. Vizzo, J. Behley, and C. Stachniss, "Rangenet++: Fast and accurate lidar semantic segmentation," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 4213–4220.

[33] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[34] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise." in *kdd*, vol. 96, no. 34, 1996, pp. 226–231.

[35] F. Dellaert, R. Roberts, V. Agrawal, A. Cunningham, C. Beall, D.-N. Ta, F. Jiang, lucacarlone, nikai, J. L. Blanco-Claraco, S. Williams, ydjian, J. Lambert, A. Melim, Z. Lv, A. Krishnan, J. Dong, G. Chen, K. Chande, balderdash devil, DiffDecisionTrees, S. An, mpaluri, E. P. Mendes, M. Bosse, A. Patel, A. Baid, P. Furgale, matthewbroadwaynavenio, and roderick koehle, "borglab/gtsam," May 2022. [Online]. Available: https://doi.org/10.5281/zenodo.5794541

[36] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard, and F. Dellaert, "isam2: Incremental smoothing and mapping using the bayes tree," *The International Journal of Robotics Research*, vol. 31, no. 2, pp. 216–235, 2012.

[37] M. Kaess, A. Ranganathan, and F. Dellaert, "isam: Incremental smoothing and mapping," *IEEE Transactions on Robotics*, vol. 24, no. 6, pp. 1365–1378, 2008.

[38] C. Qu, S. S. Shivakumar, W. Liu, and C. J. Taylor, "Llol: Low-latency odometry for spinning lidars," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 4149–4155.